1. Quy tắc css:
   1. Từ ngoài vào trong: margin , padding rồi mới đến content
   2. Từ sau ra trước: background trước
   3. Từ trên xuống dưới
   4. Từ trái sang phải
2. ul thường để display:flex
3. khối bao quanh để flex thì các khối bên trong muốn ko bị cùng hàng thì phải gom vào 1 và chỉ 1 khối con
4. Thông thường ul sẽ được để trong một khối div
5. 1 khối div chứa thẻ tiêu đề h1-h6 thì mặc định thẻ tiêu đề đã có margin-bottom với khối div >0. Do đó hoặc là bỏ thẻ tiêu đề , hoặc là set margin-bottom = 0 cho thẻ tiêu đề
6. Set background-color cho 1 khối mà chỉ set phần bao quanh nội dung thì đặt khối đó thành inline-block
7. Có thể tạo padding cho thẻ <a> bằng cách bọc nó trong một phần tử khác hoặc set nó là display:block
8. Thẻ button submit phải ở trong thẻ form
9. Khi thiết lập khoảng các cho các flex item hoặc li thì mỗi phần tử cạnh nhau phải có padding right và padding left chứ ko áp dụng padding cho chỉ 1 thằng còn thằng kế bên thì bỏ padding để thiết lập khoảng cách
10. Thẻ p , thẻ h1-h6 luôn có margin
11. Ví dụ khối ul li, các thẻ li có margin thì kích thước của khối ul ( khối cha ) sẽ bao gồm cả margin đó luôn!
12. #task-form input {
    1. padding: 5px 0px;
    2. flex-grow: 0.95;
    3. outline: none;
    4. border: none;

}

1. element form thì event là submit ; element button thì event là onclick
2. Trong DOM thì các thẻ có thuộc tính id thì tên của id đó được coi như 1 element đã lấy ra từ DOM và có thể dùng trực tiếp
3. event + functions = tương tác Js
4. img {
   1. width: 100%;
   2. /\* width: 100%; để đặt chiều rộng của hình ảnh bằng 100% chiều rộng của phần tử chứa nó. \*/
   3. height: auto;
   4. /\* height: auto; để giữ nguyên tỉ lệ khung hình ban đầu \*/
5. }
6. Để ẩn 1 khối html thì dùng display: none hoặc visibility: hidden (vẫn tồn tại trên DOM)
7. ul>li>a thì set thẻ a display:block; và style padding cho thẻ a chứ ko style cho li
8. Không set position:fixed trực tiếp cho #header vì cả layout sẽ bị sập lên trên.
9. Thay vào đó set position:fixed cho 1 khối con của #header(cụ thể là nav)

1. và sau đó set height cho #header để không bị sập độ cao
   * + Cách code responsive menu:
2. Để ul#responsive-menu position:absolute còn header hoặc nav là position:relative
3. Khi code mobile first trên PC thì cứ code cho PC trước, sau đó xem trên mobile để sửa sau (thường thì chỉ thêm display:none vào chỗ đã css đã code thôi và trên @query min-width thì display:block).
4. Ta thường set padding cho các thẻ li chứ ko set padding cho khối ul chứa nó
5. row-col-row-col : tạo danh sách sản phẩm
6. độ rộng 1 row được chia thành 12 đơn vị độ rộng column
7. Nếu cần đổi thuộc tính khi hover vào 1 phần tử thì ko được set class bootstrap về thuộc tính đó cho phần tử này
8. Cách tùy chỉnh hình ảnh với nhiều kích thước khác nhau trong 1 website để cho chúng có cùng 1 kích thước của các khối cha div-col-md của chúng

# main-img img{

Width: 100% (max-width: ?px dùng cho container responsive)

Height: 650px;